

IN THE SPECIFICATION

Page 1, line 14, please replace "\_\_\_\_\_ (Atty. Docket No. ENVSP030B)" with --08/847,816--.

Page 19, above line 20, please insert:

--In Fig. 8a, the step 108 of Fig. 7 "Activate Netscape With Plug-In" is described in greater detail. It should be noted at this point that Netscape is a convenient platform for this process 108 since it is currently the most popular Internet "browser" and because it has powerful plug-in capabilities. By "plug-in" it is meant that Netscape allows third party utilities and applications to be tightly integrated with the Netscape to enhance and expand its functionality. However, other web browsers having similar capabilities can also be used.

Therefore, process 108 will be described in terms of commercially-available Netscape Navigator software 132 that provides an interface to the Internet 18 via the Internet server 28 and a "Plug-in" 134 of the present invention which expands and enhances the capabilities of the Netscape software 132. It should be noted that the entire process 108 preferably takes place within the Internet Interface Computer 26, i.e. Netscape 132 and the plug-in 134 are preferably running on the computer 26. Also provided by a combination of hardware and software functionality on the Internet Interface Computer 26 is a voice recognition interface 136, DTMF decoding interface 138, and audio output interface 140, all of which provide interfaces between the telephone 12 and the plug-in 134, and all of which are readily commercially available.

The operation and functionality of Netscape software 132 is well-known to those skilled in the art. Also well-known is how to create plug-ins for the Netscape software. The plug-in 134 of the present invention receives a hypertext mark-up language (HTML) file 142 from Netscape 132 as part of the normal functioning of Netscape 132. A parser 144 parses the HTML file into an intermediate file 146 which is input into a converter 148. The converter uses a text-to-speech engine 150 to convert text from the intermediate file into sound, i.e. the converter 148 passes text as indicated at 152 to the text-to-speech engine 150 and receives in return, corresponding sound as indicated at 154. As noted previously, text-to-speech engines are

commercially available. The converter 148 then creates a "QuickTime Movie File" 146 which, as will be explained in greater detail subsequently, provides a convenient way for linking an audio track to hypertext links. It should be noted here that the QuickTime Movie File is an example of an audiovisual and linked hypertext engine, and that there are other commercially available engines to perform a similar function. The QuickTime Movie File 156 is then input into a player 158 which can provide audio (i.e. voice and other audio output) to the user of telephone 12 via the audio interface 140. Also, the player 158 responds to DTMF tones created by the telephone 12 and detected by the DTMF interface 138 of computer 26, and may also be responsive to the spoken word of the user into the telephone 12 via voice recognition interface 136 of the computer 26. The player 158 sends appropriate commands to Netscape 132 as indicated by arrow 160.

21  
In use, the Netscape 132 is used to make a connection to the Internet 18 via the Internet Server 28. When an HTML file is received by the Netscape 132 over the Internet, it is passed to the plug-in 134 as HTML file 142, where it is parsed by parser 144 into the intermediate file 146. The converter 148 converts text to speech and stores it in a convenient format such as a QuickTime Movie File 158. The player 158 then plays the converted HTML file as a spoken message to a user of a telephone 12 via audio interface 140. Concurrently, the player 158 is responsive to inputs and commands from the user of telephone 12 via DTMF interface 138 and/or voice recognition interface 136. If a valid command or input is received by the player 158, the player 158 will put the command or input in a proper format and send it to Netscape 132. Netscape 132 will then act upon the commands and input accordingly.

In Fig. 8b, the process of parser 144 of Fig. 9 will be described in greater detail. Process 144 begins at 162 and, in a step 164, the HTML file 142 is retrieved. Next, in a step 166, the parser process 144 reads from the HTML file and, in a step 168, parses the HTML file text using a Perdue Compiler Construction Tool Set (PCCTS) or Yet Another Compiler Compiler (YACC). Both PCCTS and YACC are commercially available. Next, in a step 170, it is determined whether the results of the parsing is "user displayable" text. For example, the parsing of step 168 might find some "user displayable" text intermixed with graphics, photographs, Java Applets, etc. which, in the present preferred embodiment, would not be considered user

21  
displayable text. If step 170 determines that there is user displayable text, a step 172 "writes" the user displayable text to the intermediate file 146. If step 170 determines that it is not user displayable text, or after step 172 has been completed, a step 174 determines if there is a hyperlink embedded in the HTML file 142. If there is a hyperlink, it is written to intermediate file 146. If step 174 determines there is no hyperlink, or after the completion of step 176 if there is a hyperlink, a decision step 178 determines if an end of file (EOF) has been read from the HTML file 142. If not, process control is returned to step 166 to continue reading the HTML file. If step does 178 does detect the EOF of the HTML file 142, the process 144 is completed as indicated at 180.--

---

22  
Page 19, line 20, before sentence beginning with the phrase "Fig. 9 is a visual" please insert: --The parsing process 144 will be explained by way of example with Figs. 9 and 9a.--

---

23  
Page 20, before line 3, please insert:

---

-- Fig. 9a illustrates the intermediate file 146 of Fig. 8a that has been created from the HTML file 142 by the parser 144. As noted in Fig. 9a, the text 164 has been parsed as text, the hyperlinks 166-170 have been parsed into hyperlinks and text 178, text 176 has been parsed as text, and the hyperlink 172 has been parsed as hyperlink and text 180. Unrecognized objects, such as non-text object 74, is not added to the intermediate file 146. Intermediate file 146 is then ready for conversion from text-to-speech by converter 148.--

---

24  
Page 24, before line 25, please insert:

---

-- In Fig. 16, the method of converter 148 is illustrated. The method 148 begins at 142, and in a step 184, the intermediate file 146 is retrieved. Next, in a step 186, a segment of the intermediate file is read. This portion of the intermediate file is parsed again, preferably, using PCCTS in a step 188 and, in a step 190, it is

determined whether there is user displayable text. If there is user displayable text, a step 192 converts text to speech and other sounds. For example, special sounds such as "siren" could be converted either to the sound of the siren or to the spoken word "siren." If a word was to be converted into a sound, rather into a spoken word, a special identifier such as <sound> can be used as a preference tend to be used as an indicator. For example, "<sound>siren" would make the siren sound, while "sound" alone would be read as "sound." Next, in a step 194, the sound is written to a QuickTime Movie File or the like.

If step 190 determines that user displayable text is not present, or after the completion of step 194, a decision step 196 determines whether there is a hyperlink. If yes, a step 198 converts the text portion to speech sound, the sound is written to a quick time file in a step 200, and the URL is written to the QuickTime Movie File in a step 202. If step 196 determines that there is not a hyperlink, or after the completion in step 202, a decision step 204 determines whether an end of file (EOF) has been found in the intermediate file 146. If not, process control is returned to step 186. If an EOF has been found, the process is completed as indicated at 206.

In Fig. 16a, a QuickTime Movie File 156 is illustrated. As noted previously, there are alternatives for QuickTime Movie File which provide similar functionality. The QuickTime Movie format operates on Macintosh platforms, and there are equivalent utilities available for other platforms such as "WINTEL" platforms running Intel-type microprocessors and Microsoft Windows operating system software. In the illustration of Fig. 16a, the QuickTime Movie File includes an audio track 208 and a "metadata" track 210. Time increases from left to right in this illustration as indicated by arrow 212. At irregular intervals, the audio track is linked to a hyperlink in the metadata track 210, such as with hyperlinks 214, 216, and 218. These hyperlinks were written by step 202 of Fig. 16. Each hyperlink 214-218 is given a reference number and a URL. For the purpose of this illustration, the entire URL has not been written. Also, it should be noted that the reference number can be reassigned, such as with hyperlink 218 having the reference number 4 which was reassigned from hyperlink 214. Reassignment is needed because there are many more possible hyperlinks than telephone keypad keys. This QuickTime Movie File 156 therefore provides a convenient method for linking an audio track with a metadata track including a

number of hyperlinks. The use and functionality of QuickTime Movie File are described in greater detail in the multi volume set Inside Macintosh, Addison-Wesley Publishing Company, Inc.

In Fig. 17, the process of the plug-in 134, under the control of the player 158, will be described. The process of 134 begins at 220 and, in a step 222, the player 158 commands Netscape to go to the desired home page, i.e. the starting home page. Next, in a step 224, the HTML file 142 is received from Netscape 132 and, in a step 226, the parser 144 converts the HTML file 142 to the intermediate file 146. Step 228 converts intermediate file to a QuickTime Movie File 156 with the converter 148, and the player 158 starts playing the audio track of the QuickTime Movie File in a step 230. The player 158 also processes user commands retrieved through DTW interface 138 or voice recognition interface 136 in a step 232, and the process for that HTML file is completed at 234. Alternatively, as indicated at 236, an additional HTML file can be retrieved from Netscape in a step 224 and the process can be repeated for this new HTML file.

In Fig. 18, the step 232 of Fig. 17 "Process User Command" is shown in greater detail. The process 232 begins at 238 and, in a step 240, a command from the DTMF interface 138 or the voice recognition interface 136 is received. If the command is "Rewind" as determined by a step 242, QuickTime Movie File 156 is "rewound" to the beginning of the audio track in a step 244. If the command is "Forward 10 Seconds" as indicated by step 246, the audio track on the QuickTime Movie File 156 is advanced by 10 seconds. If the command is to "Go Back 10 Seconds" as detected by step 250, the audio track is "rewound" 10 seconds on the QuickTime Movie File 156. If a step 154 indicates that the user wishes to jump to a hyperlink, the URL is looked up in a step 256 and a command is created in a step 258 for Netscape 132 to jump to the appropriate URL. Process control is then returned to step 224 of Fig. 17 to receive a new HTML file from Netscape. If step 260 determines that the user is done (for example by receiving a "done" command from the user or by sensing that the user has hung-up the telephone), the process is completed as indicated at 262. If the telephone user is not done, process control is returned to step 240 to retrieve a new command from the user of telephone 12.